

Towards a Railway Topology Ontology to Integrate and Query Rail Data Silos

Stefan Bischof^[0000-0001-9521-8907] and Gottfried Schenner^[0000-0003-0096-6780]

Siemens AG Österreich, Corporate Technology, Vienna, Austria
bischof.stefan@siemens.com, gottfried.schenner@siemens.com

Abstract Engineering projects in the railway domain typically involve a large number of subsystems. Therefore a common understanding of the domain is essential. In the past this has been provided by XML-based data exchange standards and UML-based object-oriented models. With the increasing adoption of semantic technologies for engineering projects the demand to provide these standard data models in the form of ontologies has grown. We describe requirements and challenges to define an open standard ontology for railway topologies based on existing standards. The purpose of the finished ontology will be to enable topological queries and reasoning for railway networks in a standardized and reusable manner.

Keywords: industrial knowledge graph · ontology · railway

This is an ISWC 2020 poster submission.

1 Introduction

A common understanding of the different railway subsystems is vital for successful railway engineering projects. Historically, plans, drawings and lists were used to exchange information for engineering between stakeholders. Digital equivalents replaced these artifacts. To simplify railway simulation and operation applications, data exchange formats were standardized. Ideally, we could simply integrate rail data exports and thereby enable a domain expert to query data from a wide array of systems in an integrated knowledge graph system.

Different rail infrastructure systems and subsystems are modelled by different tools resulting in a number of *independent* files describing the same real-world station. Automatically integrating these files is hardly feasible due to different modelling approaches and the lack of a canonical naming scheme of all different entities. To pose a query to a system with integrated data, domain experts have to know the domain model of the system well. Introducing yet-another domain model puts system adoption at risk.

We start to address these two problems with a common unified railway topology, formalised as an OWL ontology. RDF and OWL together with an

Copyright © 2020 for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

ETL (extract, transform, load) pipeline should simplify the data integration task and international standards as a base should increase adoption [3]. In this poster we discuss different existing railway topology models, formulate concrete requirements and describe a preliminary ontology to enable certain topological queries.

2 Requirements

Our ontology engineering process is guided by three resources: competency questions, adherence to data exchange standards and best practices.

In a first iteration the following two competency questions were chosen:

1. If a train runs from A to B on the railway network, which infrastructure elements (including their direction and orientation) will be traversed?
2. What are the possible paths between A and B on the railway network?

We have the following requirements regarding modelling the railway domain and adherence to existing data exchange standards: (i) The ontology must be able to represent the (logical) topology of a railway network, (ii) The ontology must at least contain the main infrastructure elements found in every railway network, i.e., track, switches, signal and (iii) The (logical) position and orientation of these infrastructure elements in the railway network must be defined.

When designing the railway ontology we are following ontology engineering best practices as far as possible. Specifically, the ontology should be vendor independent, reusable and openly available. The concepts of the ontology must be well documented (especially important in engineering as it must be absolutely clear which concepts of the real world correspond to concepts of the ontology) and the ontology should be minimal, i.e., does not contain any aspects not related to the topology of railway networks.

3 Existing Models of Railway Topology

Historically, many different models for railway systems topology have been proposed, starting with straightforward graph-based models, for example by Hansen [6]. Figure 1 shows how a section of an example railway network (shown at the top) consisting of three signals, three tracks and one switch would be represented in three different models:

The *component-port model* – here every relevant infrastructure element is represented by a component which is connected to other elements by ports. The EURIS method [4] uses this model.

The *double vertex graph model* – which is an extension of a standard graph-based model. Instead of a single vertex, double vertices are used, each vertex representing one possible direction of a vehicle on the railway network. Commercial tools like the railway network simulation tool OpenTrack [8] use this model.

The *connectivity graph model* turns earlier ideas upside down: instead of crossings and switches being modelled as nodes, the connectivity graph [5] models the tracks

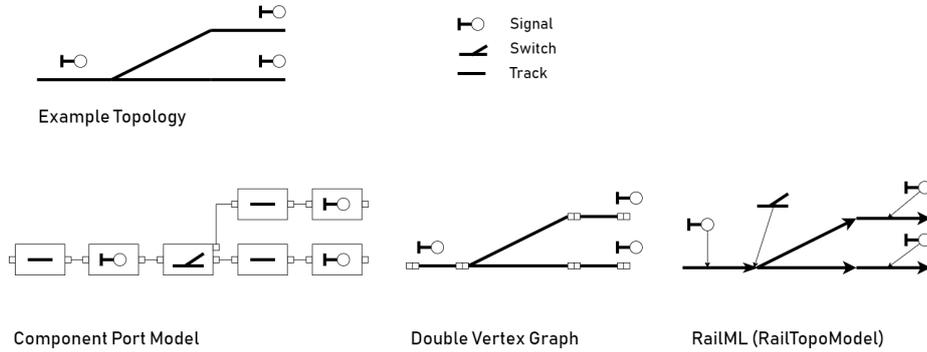


Figure 1. Railway network topology models

in between as nodes called `NetElements` connected by `NetRelations`. `NetElements` and `NetRelations` can furthermore be recursively composed on different levels of abstraction. `RailTopoModel` (RTM) [1] uses the connectivity graph model to represent the railway network topology. We decided to base our ontology on the `RailTopoModel` because it is a standard of the International Union of Railways (UIC).

4 Rail Topology Ontology

None of the earlier existing work with comparable goals like `RaCoOn` [10], `InteGRail` [9] or `Smart-Rails`¹ satisfied our requirements completely. Common issues are limitations to a small number of use-cases, a single vendor, a different modeling scope or simply missing ontology resources. General transportation ontologies like `km4city`² contain railway concepts but are too unspecific to answer the topological questions required in an railway engineering context.

To avoid some of these limitations, ensure adoption and avoid creating yet-another rail domain model, we reuse existing standards as far as possible. For our purposes and requirements the most relevant standards are RTM and RailML. RTM and RailML share a similar goal with our approach to “foster the Federation of Railway Digital Models” [7] (RTM) and reducing the number of system-to-system interfaces and data exchange formats (RailML). RTM is an abstract topology data model instantiated by RailML. RailML is an XML based data exchange format for railway data.

Although RailML provides XML-Schema files, automatically generated ontologies turned out impractical and needed considerable manual effort to be practically usable. For example it is unclear if XSD element types correspond to classes of the ontology or are merely a construct for structuring the XML

¹ <https://ontology.tno.nl/smart-rail/>

² <http://www.disit.org/km4city/schema>

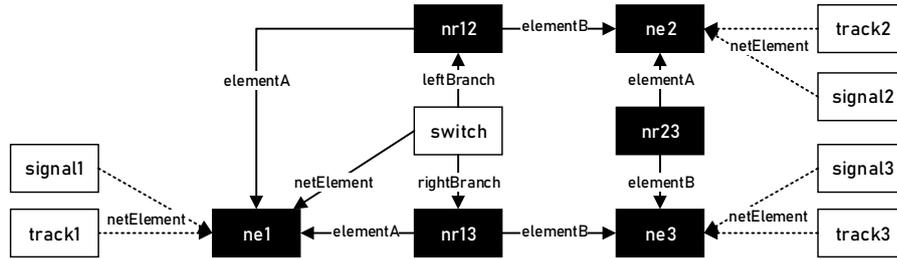


Figure 2. Instance model consisting of `topology` and `infrastructure` of the simple example in Fig. 1. Dotted connections hide corresponding nodes for positioning functional infrastructure entities on the topology.

file. Similar problems, although to a lesser extent, occurred when automatically transforming the RailML UML model to an ontology.

We expect the number of concepts and properties of a usable ontology to be relatively small, thus we use a bottom up approach, i.e., add those concepts that are mandatory for satisfying the requirements of our envisioned ontology.

Figure 2 models the example topology of Fig. 1 by using the adapted concepts of RTM and RailML. On the topological level, based on the RTM, `NetElements` (`ne1`, `ne2`, `ne3`) are connected to each other by `NetRelations` (`nr12`, `nr13`, `nr23`). Attributes define how ends of `NetElements`, e.g., ends of a track, are connected. Functional infrastructure entities from RailML, tracks, signals and switches, are then attached to the topology entities and thereby located.

5 Conclusion

Knowledge Graphs and Semantic Web technologies are promising technologies to integrate rail data from different systems. The graph nature of rail networks together with a large number of highly interconnected parts and components suits graph-based representations well.

XML standards such as RailML have been created to mitigate the proliferation of schemas and object-oriented models for a domain, i.e., to at least allow a tool-independent exchange of data. Building on top of these efforts and together with the RailML community, we are currently in the process of defining a reusable standard ontology for rail topology and specific parts of rail infrastructure to support engineering. This is an ongoing effort and we invite everybody interested to contact us via email or the RailML forum. We also plan to contact research groups that have done similar work in the past, e.g., the RAISO ontology [2].

In a typical engineering tool the data of a RailML file is converted into a tool-specific data schema. This conversion step is no longer needed in a Knowledge Graph system as the instance data based on the RailML ontology can be imported directly and linked to additional data sources. This enables the reuse of queries and algorithms based on the ontology.

Furthermore, we are currently working on mappings from existing proprietary infrastructure data to the presented ontology as well as an implementation to answer our competency questions.

While some features of SPARQL 1.1, such as property path expressions, are powerful tools to explore an RDF graph, it is not clear yet if, or to what extent, our competency questions can be answered by a SPARQL endpoint with OWL reasoning alone – extensive postprocessing might be necessary.

To guide future developments, we are working with different user groups to formulate more competency questions relevant for their business.

References

1. RailTopoModel – railway infrastructure topological model. Standard, International Railway Solution IRS 30100:2016, International Union of Railway (UIC) (2016)
2. Bellini, P., Nesi, P., Zaza, I.: RAISO: Railway infrastructures and signaling ontology for configuration management, verification and validation. In: 2016 IEEE Tenth International Conference on Semantic Computing. pp. 350–353. IEEE (2016)
3. Bischof, S., Schenner, G.: Challenges of constructing a railway knowledge graph. In: The Semantic Web: ESWC 2019 Satellite Events. pp. 253–256 (2019)
4. van Dijk, F., Fokkink, W., Kolk, G., van de Ven, P., van Vlijmen, B.: EURIS, a specification method for distributed interlockings. In: SAFECOMP. pp. 296–305 (1998)
5. Gély, L., Dessagne, G., Pesneau, P., Vanderbeck, F.: A multi scalable model based on a connexity graph representation. *Computers in Railways XII* **1**, 193–204 (2010)
6. Hansen, K.M.: Formalising railway interlocking systems. In: Nordic Seminar on Dependable Computing Systems. pp. 83–94. Citeseer (1998)
7. Magnien, A.: RailTopoModel – a cornerstone to foster the federation of railway digital models. In: RSSRail’19 (2019)
8. Nash, A., Huerlimann, D.: Railroad simulation using opentrack. *WIT Transactions on The Built Environment* **74** (2004)
9. Shingler, R., Fadin, G., Umiliacchi, P.: From rcm to predictive maintenance: The integrail approach. In: 2008 4th IET International Conference on Railway Condition Monitoring. IET (2008)
10. Tutchter, J., Easton, J.M., Roberts, C.: Enabling data integration in the rail industry using rdf and owl: The racoon ontology. *ASCE-ASME Journal of Risk and Uncertainty in Engineering Systems, Part A: Civil Engineering* **3**(2), F4015001 (2015)